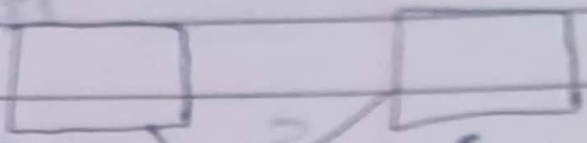
Objective of Security

CIA
- Confidentiality
- Integrity
- Availability

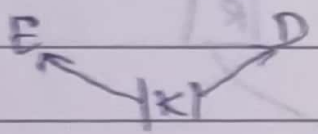


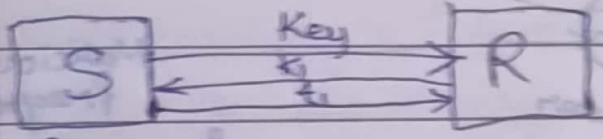
Symmetric Cryptography

→ Same key for encryption & decryption

C_x → Credit Card no. (Plain Text)

$E(C_x) = t$ Encryption Algo





C_x

$E(C_x)_{k_1} = t_1$

t_1 (Cipher text)

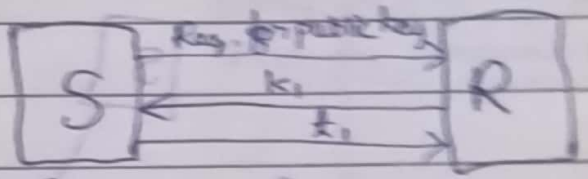
$D(t_1)_{k_1} = C_x$

Asymmetric Cryptography

→ Different key for encryption & decryption.

Public Key — K_1 (Encryption)

Private Key — K_2 (Decryption)



C_x

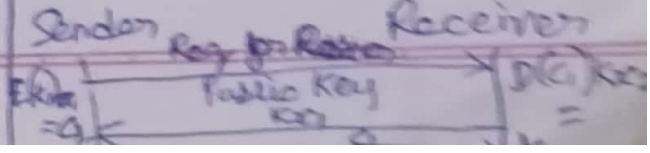
$E(C_x)_{K_1} = t_1$

K_1 → Public Key

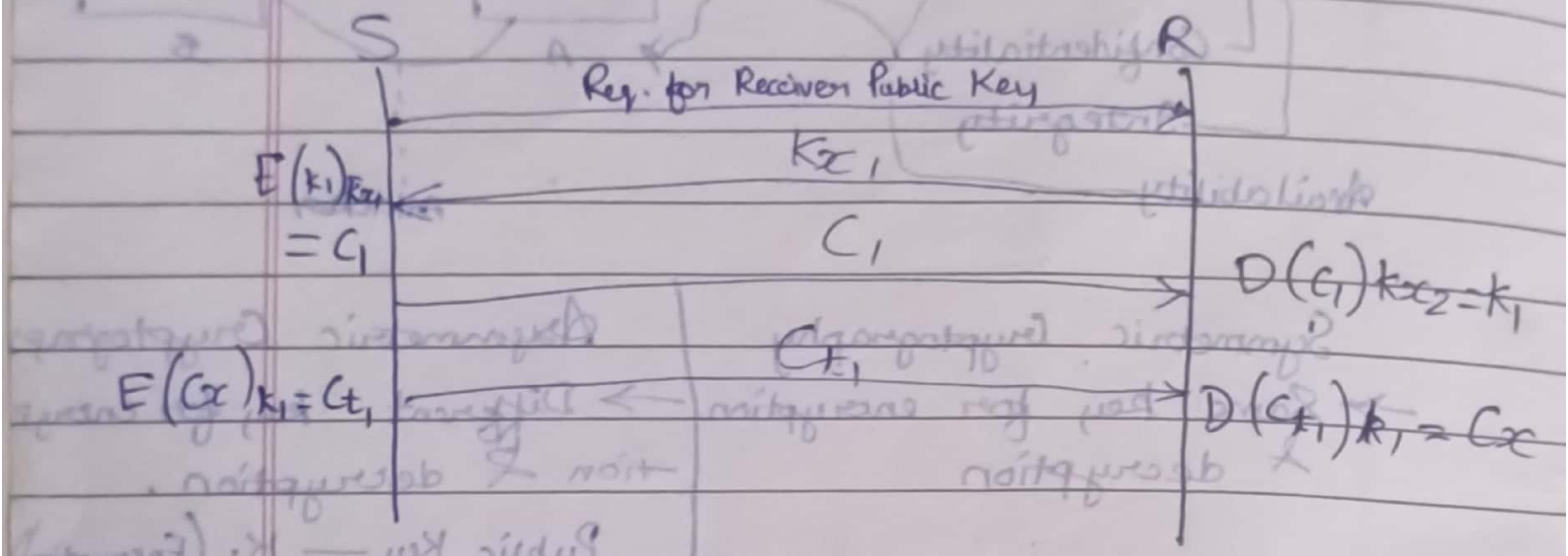
K_2 → Private Key

$D(t_1)_{K_2} = C_x$

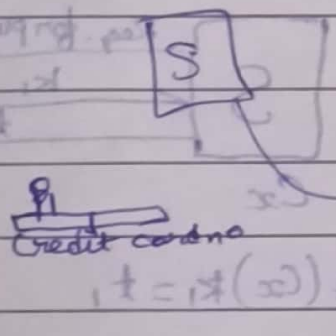
Exchange K_1 (Symmetric Key) in Secured Manner over communication channel.

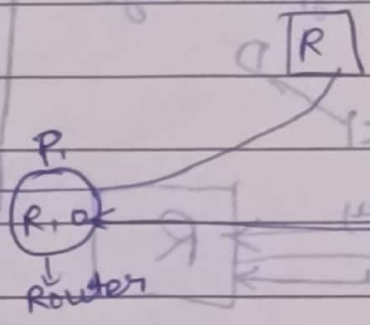


Algo : Kx_1 → Public Key
Kx_2 → Private Key



Integrity





$dx_1 - dx = 0$
then message accepted
else rejected

P_1 → Credit Card no.

$h(P_1) = dx$

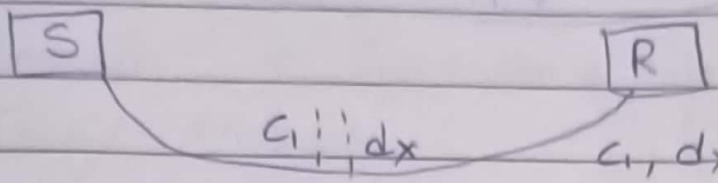
P_1 → $h(P_1)$ → dx

P_1 → dx → old message digest
$h(P_1) = dx_1$ = new message digest

Algo
MD4, MD5, SHA 1, SHA-256
SHA-512, SHA 3

128 bit

* Integrity corresponding Hashing Technique



Confidentiality: AES → Encryption / Decryption / both

$D(C_1)_{K_1} = P_1$

$h(P_1) = dx_1$

$dx_1 - dx = 0$

If this happens, this means no one has modified the message over the communication channel.

Hashing: MD5 / SHA 1, SHA 256

$E(P_1)_{K_1} = C_1$ → Encryption

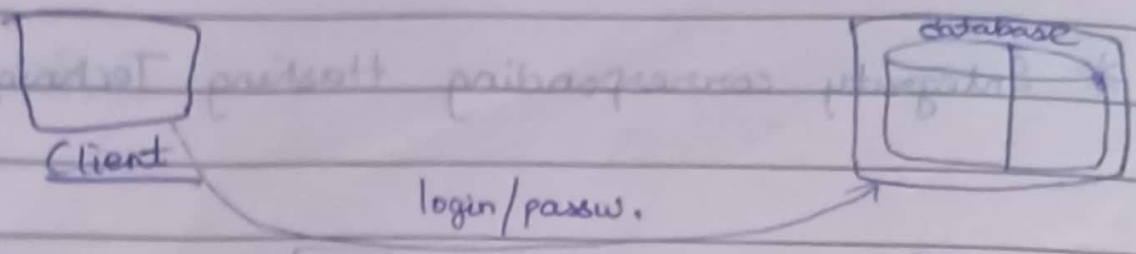
$h(P_1) = dx$ → Hashing Technique

Availability: Information should be available to legitimate user only.

Authentication: Process of verifying user credential at receiver's end.

Authorization

- ACL (Access Control List)
- ACM (Access Control Matrix)



- ACL → Access Control List
  Database — Object

User	read(r)	write(w)	Execute (x)	Append (a)
Bank Manager	✓	✓	✓	✗
Admin	✓	✗	✗	✗
Security Expert/DB Admin	✓	✓	✓	✓
Bank Clerk	✓	✗	✗	✗
Cashier	✓	✓	✗	✗

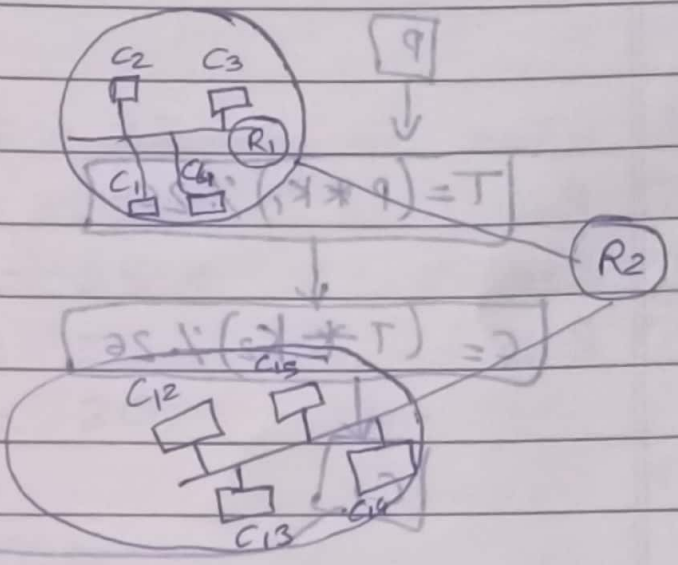
- ACM → Access Control Matrix

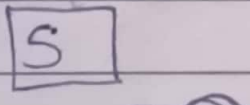
	Bank Manager	Admin
Database	r w x	r - -

implement Ceasar Cypher Technique

* Threats to Confidentiality
  - Snooping — Interception of data from communication channel
  - Traffic Analysis

* Threats to Integrity
  - Modification
  - Masqueration
  - Replay
  - Repudation





* Attacks on Unavailability
  - Denial of Service (DoS)

$C_1 = (23+2) \% 26 = 25$

$C_2 = (2+2) \% 26 = 4$

$C_3 = (25+2) \% 26 = 1$

$C_4 = (25+2) \% 26 = 1$

$C_5 = (20+2) \% 26 = 22$

Cipher Text = zebbw

Decryption

$T = (25-2) \% 26 = 23$

$(4-2) \% 26 = 2$

$(1-2) \% 26 = 25$

$(1-2) \% 26 = 25$

$(22-2) \% 26 = 20$

$P = (23 * 7^{-1}) \% 26$

Q. Using affine cipher tech., convert following text cipher text & vice-versa.

P. Text = COLLEGE KEY = (17, 20)

K_1 K_2

C = 2 E = 4

O = 14 G = 6

L = 11 E = 4

L = 11

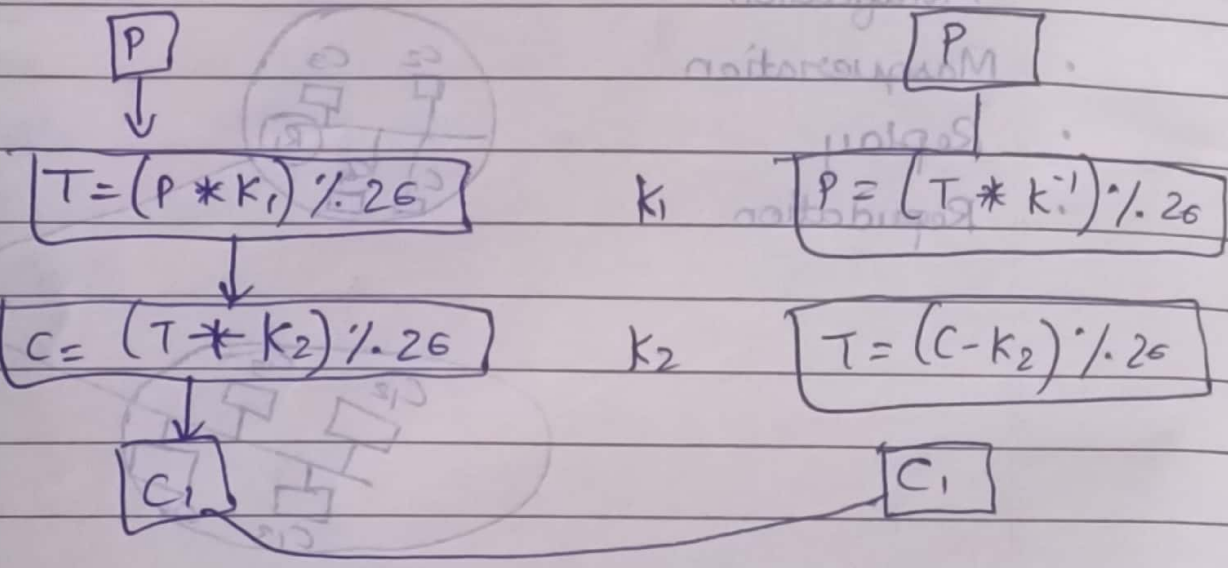
* Monoalphabetic Cipher Technique

(a) Additive Cipher Tech

P

$C = (P + K) \% 26$

* Affine Cipher : Combo. of multiplicative & additive cipher



* Encrypt and decrypt "Hello" using affine cipher

Key = (7, 2)

Ans

h = 7	⟶	(7 * 7) % 26 = 23
e = 4		(7 * 4) % 26 = 2
l = 11		(7 * 11) % 26 = 25
l = 11		(7 * 11) % 26 = 25
O = 14		(7 * 14) % 26 = 20

$T_i = (P * K_1) \% 26$	
$= (2 * 17) \% 26 = 8$	$(8+20) \% 26 = 2 = C$
$= (14 * 17) \% 26 = 6$	$(4+20) \% 26 = 24 = Y$
$= (11 * 17) \% 26 = 5$	$(5+20) \% 26 = 25 = Z$
$= (11 * 17) \% 26 = 5$	$(5+20) \% 26 = 25 = Z$
$= (4 * 17) \% 26 = 16$	$(16+20) \% 26 = 10 = K$
$= (6 * 17) \% 26 = 24$	$(24+20) \% 26 = 18 = S$
$= (4 * 17) \% 26 = 16$	$(16+20) \% 26 = 10 = K$

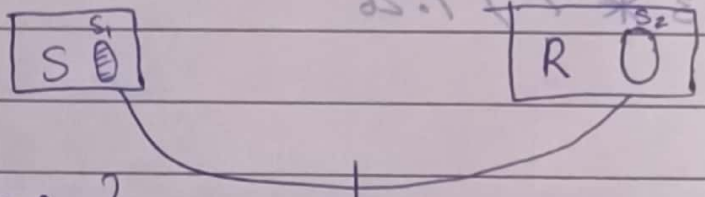
* When no key is given.. Then find

① Rule 1: $\gcd(K_1, 26) = 1$

② Rule 2: $1 \leq K_1 \leq 25$

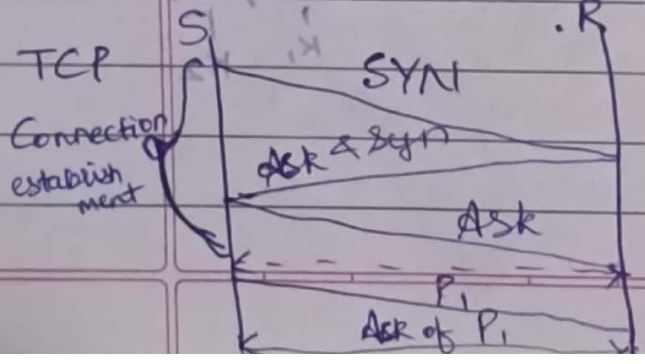
③ Rule 3: $0 \leq K_2 \leq 25$

* Protocol Level Vulnerability



TCP
- SDP
} Transport Layer

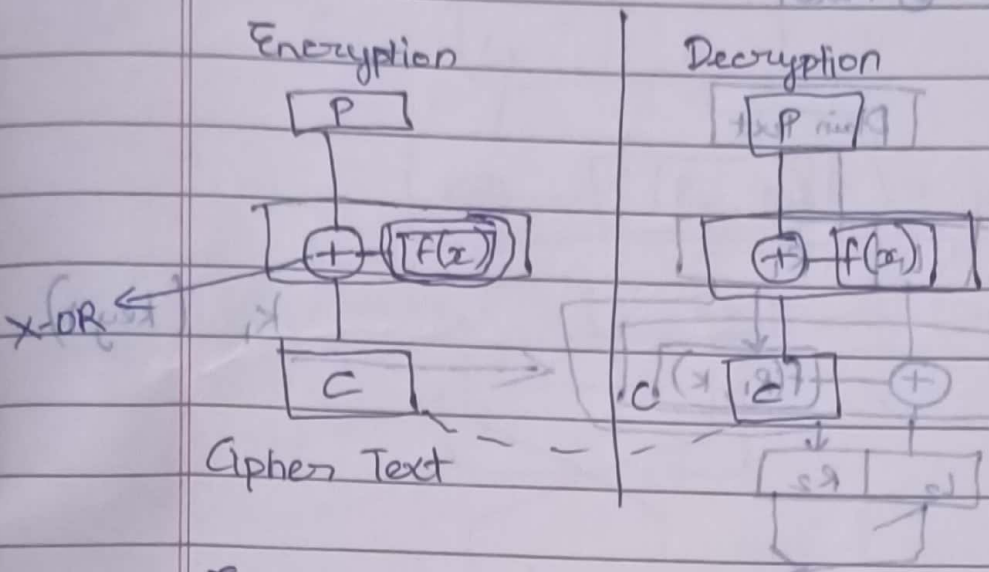
TCP is a connection oriented reliable protocol.



Resource

TCP - Transmission Control Protocol.

① First Draft of Feistal Cipher — In first draft the X-OR operation works only on the Key.

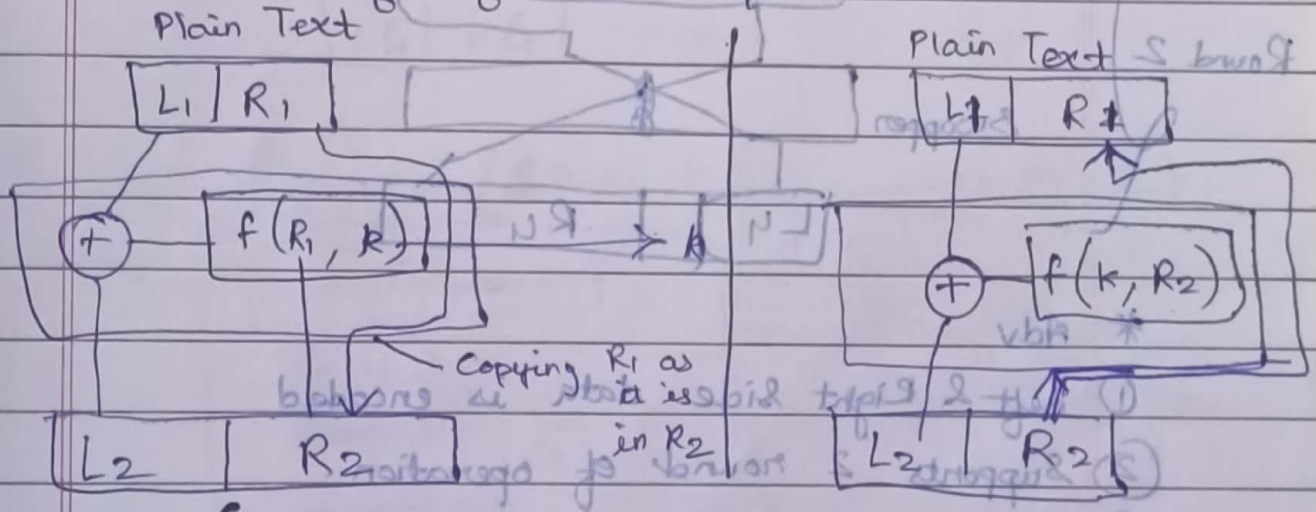


Ex P = 00 K = 01

$C = P_1 \oplus f(k) = 00 \oplus 01 = 01$

$P = 01 \oplus 01 = 00$

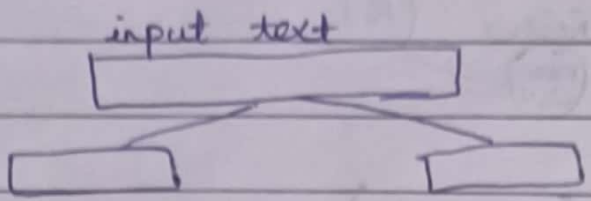
② Second Draft of Feistal Cipher



In this draft, no change is made in Right side data so it is in plain text format.

PAGE No.
DATE / /

Block Cipher

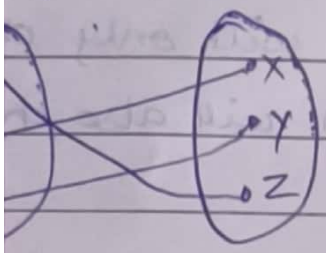


DES ② AES algos.

Feisal Cipher Technique

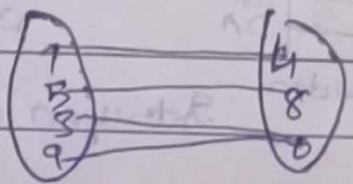
① Invertible ↓

Every input has unique output



② Non-invertible ↓

Two inputs can have same output

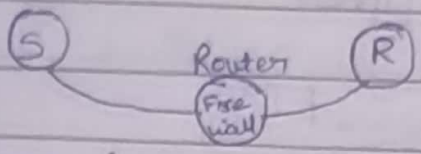


③ Self Invertible ↓

Function is equa to its inverse.

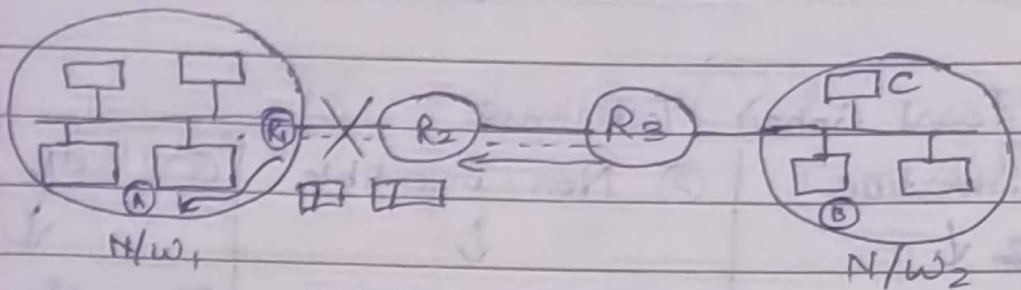
$f(x) = f^{-1}(x)$

Feisal Cipher Techniques uses X-OR operation.
X-OR doesn't loose any information during encrypti or decryption.
It is reversible.
It induces randomness in the algorithm.
X-OR function cancels the effect of encryption during decryption operation.

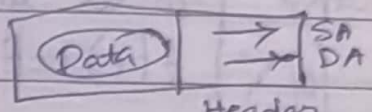


IP Spoofing

(ICMP Flooding) — DoS attack



ICMP packet is rarely used.



Firewall will only check Header part.
Intrusion detection will also check the data part.

Stream Cipher:

Playfair Cipher Key = moon

① 5 * 5 mtx.

m	o	n	a	b
c	d	e	f	g
h	i/j	k	l	p
q	r	s	t	u
v	w	x	y	z

① ab cd ef — Form Pairs

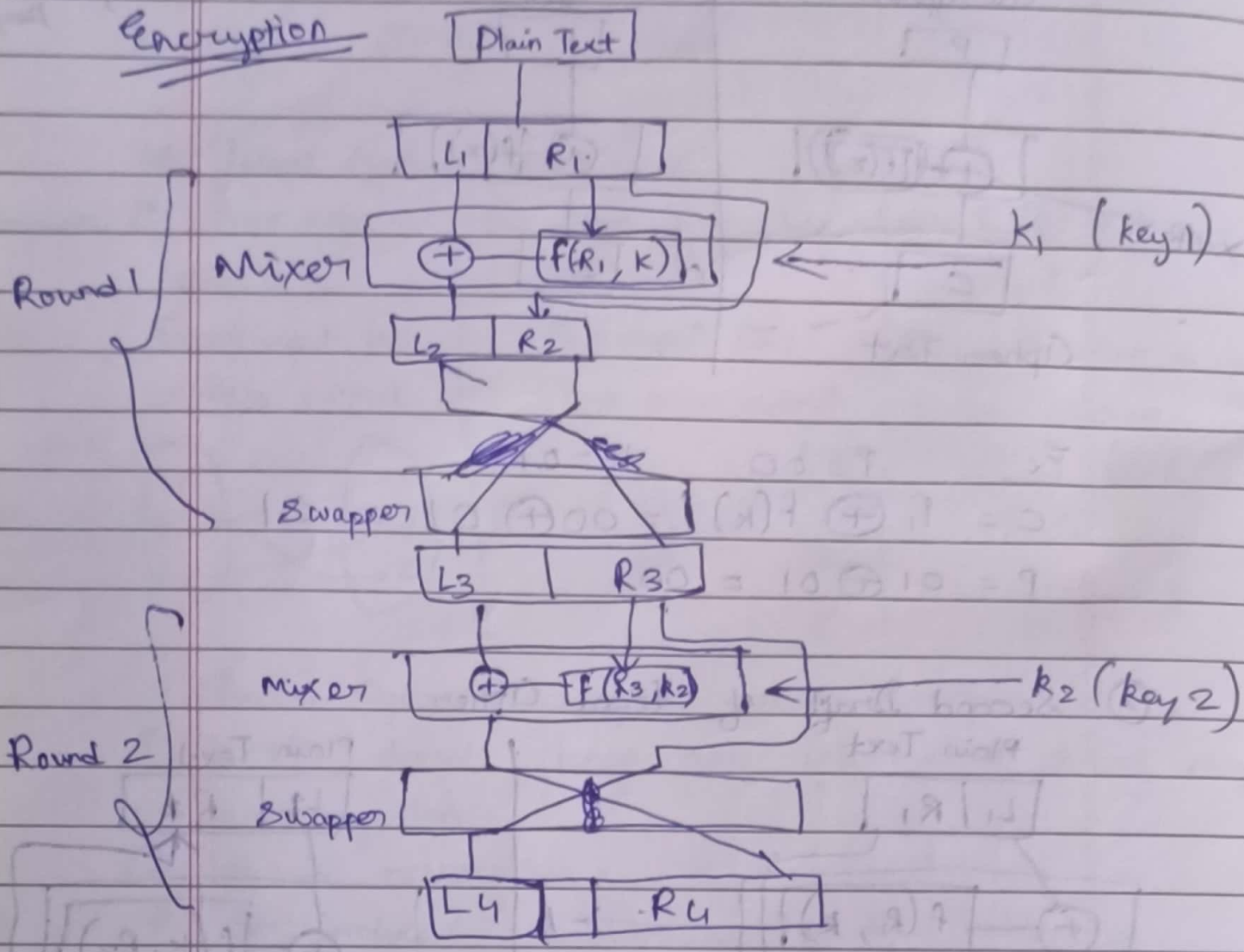
② atty → Repeat
hua to beech no stuff x.

③ ab cd ex → Empty raha to stuff x.

PAGE No.
DATE / /

3rd Draft of Feissal Apher ~~Comp~~ Technique

① Swapper ② Mixer



* Adv

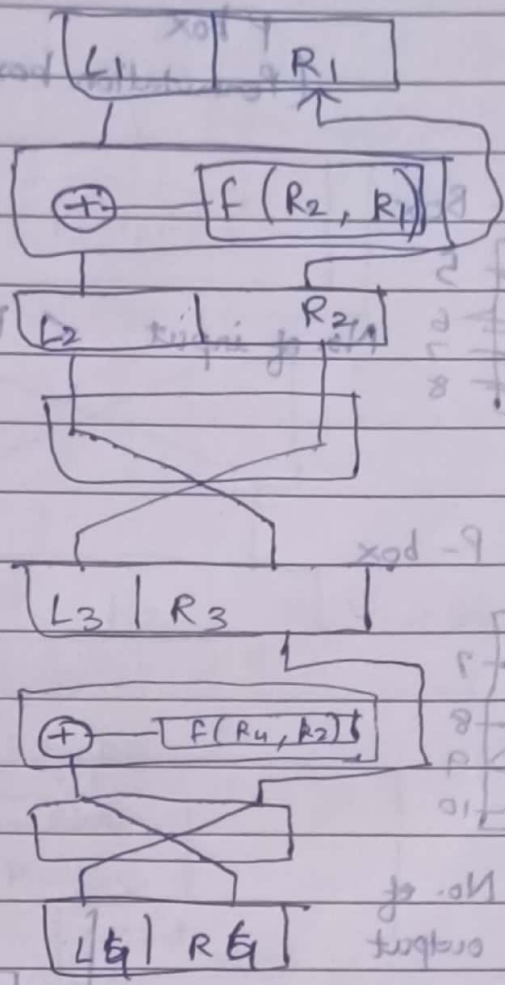
① Left & Right side data is encoded

② Supports 2 round of operation.

③ Along with mixer, swapper is also operated.

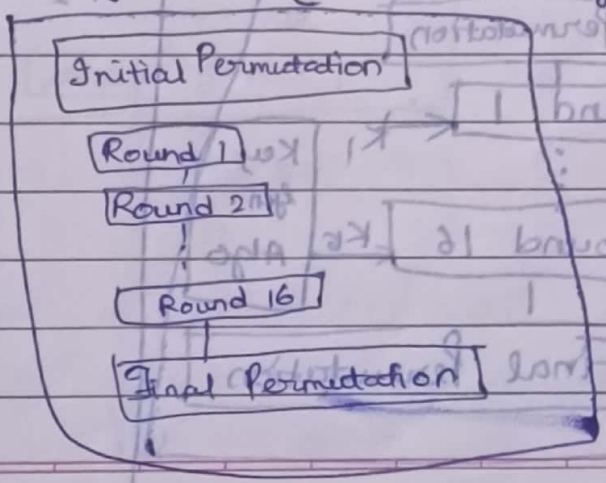
④ 2 keys used for enc, decr

Decryption

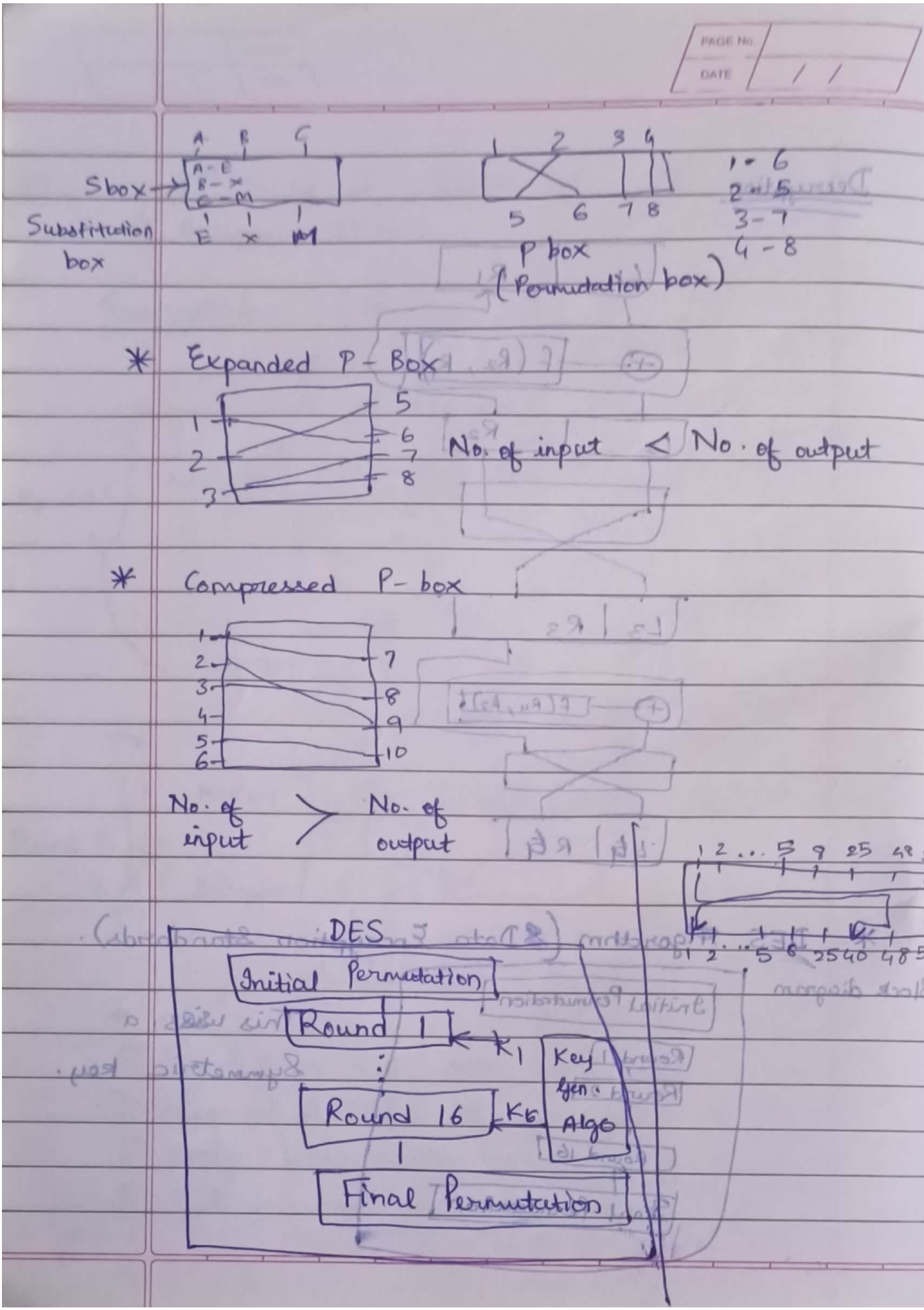


* DES Algorithm (Data Encryption Standards).

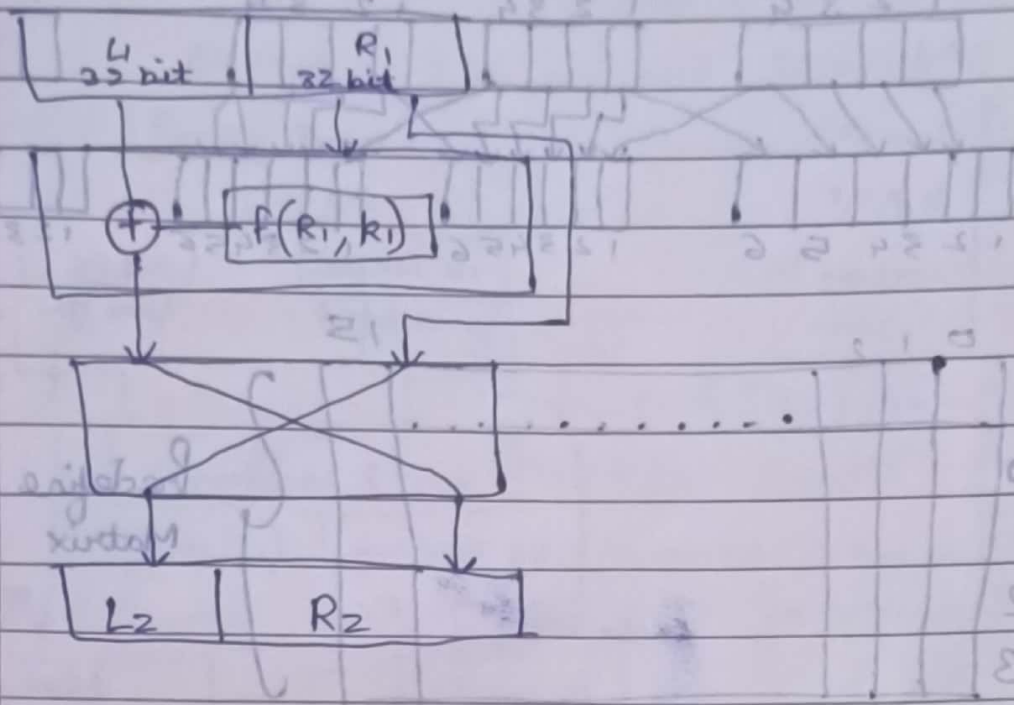
Block diagram

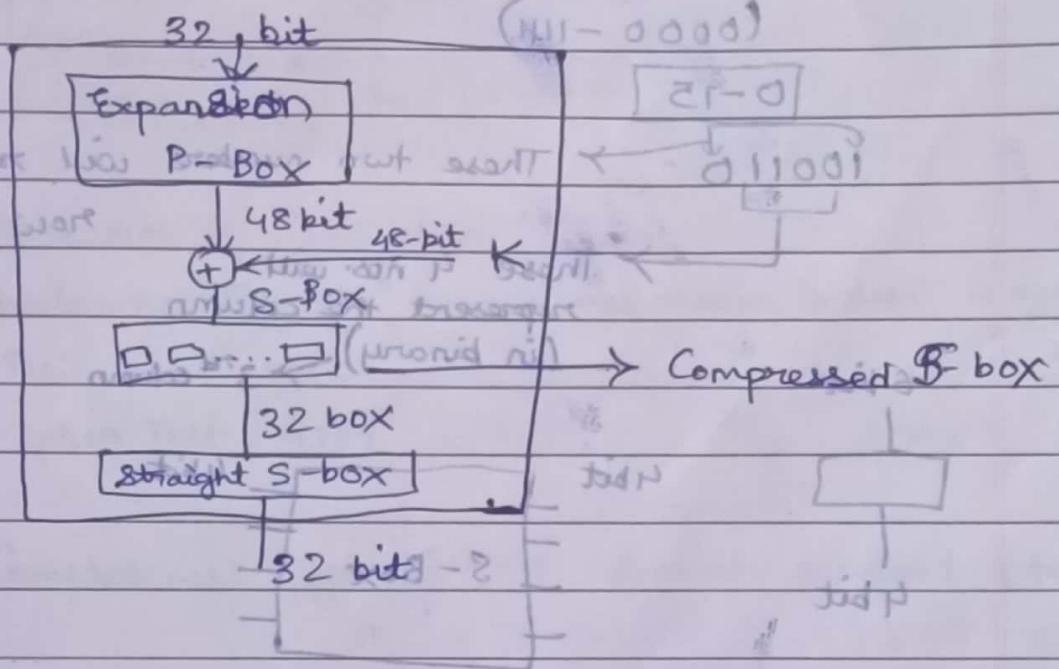


This is a Symmetric key.

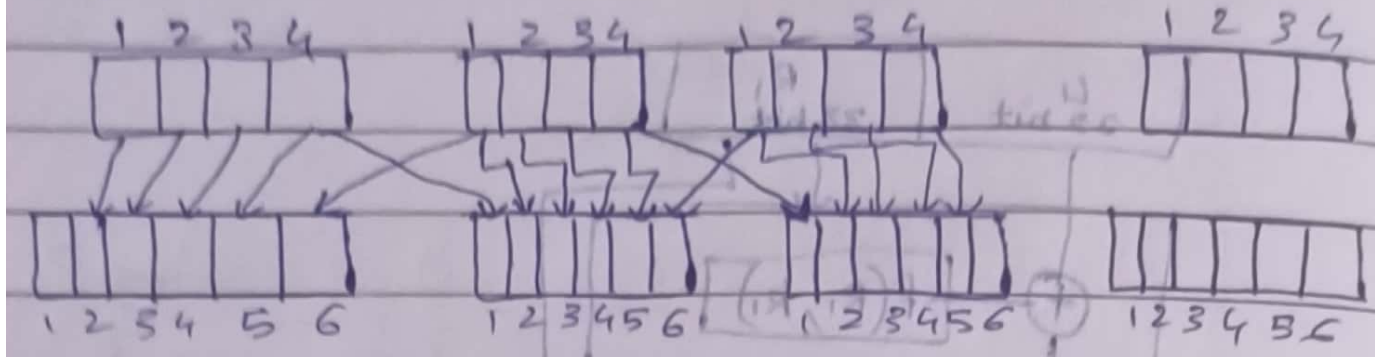

A B C
A - E
B - X
C - M
Sbox
Substitution box
E X M
1 2 3 4
5 6 7 8
1 - 6
2 - 5
3 - 7
4 - 8
P box
(Permutation box)
* Expanded P - Box
1 2 3
5 6 7 8
No. of input < No. of output
* Compressed P - box
1 2 3 4 5 6
7 8 9 10
No. of input > No. of output
1 2 ... 5 9 25 48 58 64
1 2 ... 5 6 25 40 48 58 64
DES
Initial Permutation
Round 1
K1
Round 16
K16
Key Gen. Algo
Final Permutation

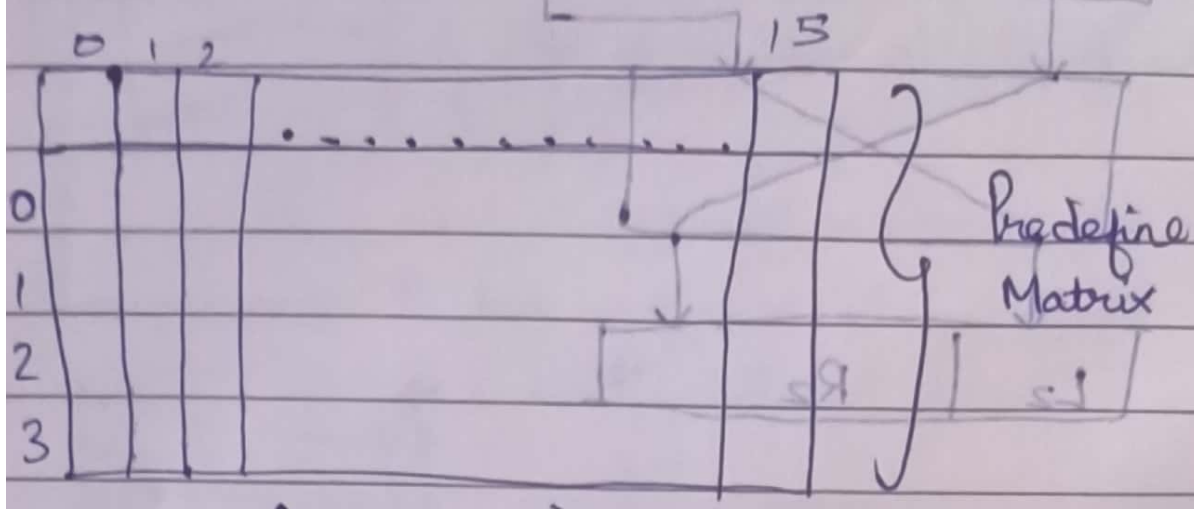
* Round 1 operation





Expansion P-box



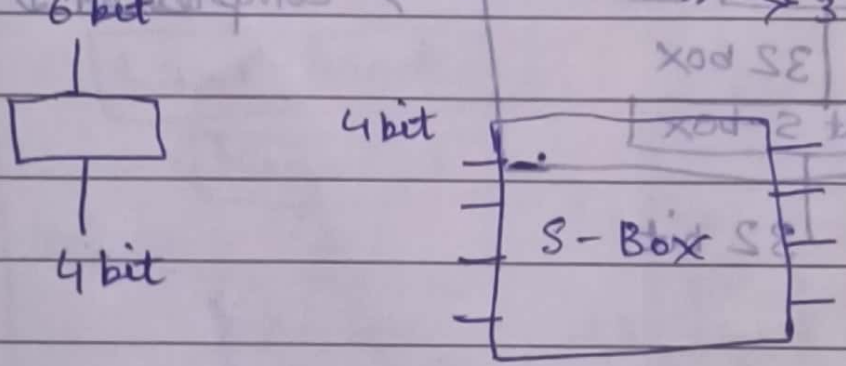


(0000 - 1111)

0-15

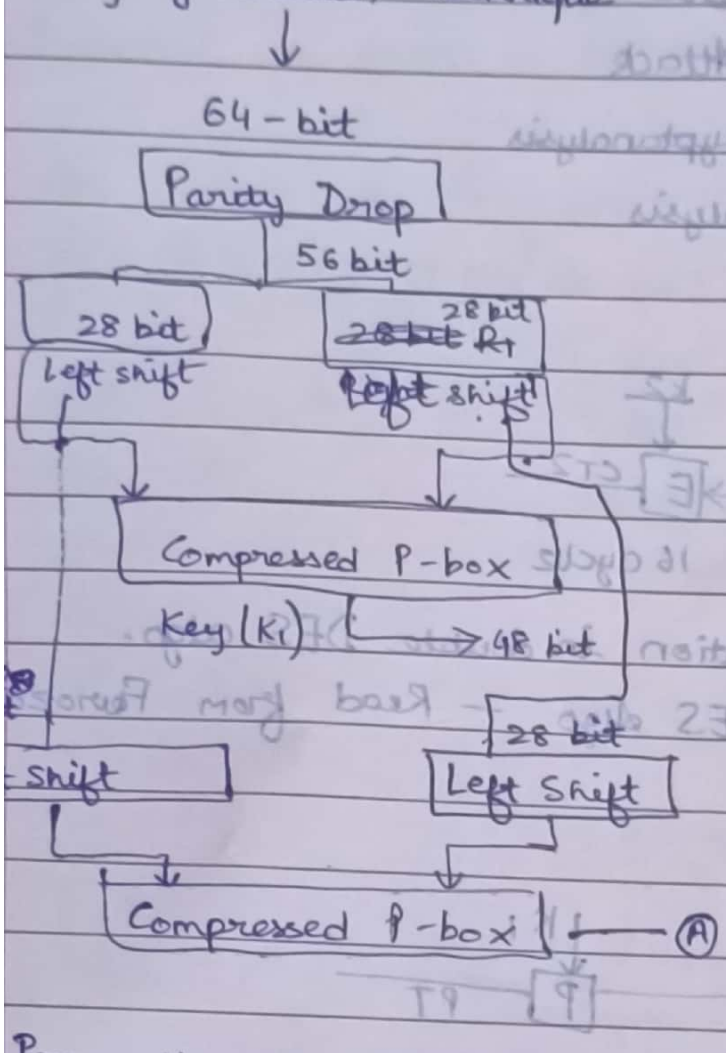
100110 → These two numbers will represent the row (in binary) i.e. 10 → 2nd row

These 4 nos. will represent the column (in binary) → 3rd column



In S-box, input character is replaced by its corresponding output character as per the table designed by algorithm or formula.

Key Generation Technique



Round	Shift
1, 2, 9, 16	1 bit LS
Others	2 bit LS (Left shift).

Properties of DES Algo.

Avalanche Effect : Even a small change in data, it causes a huge change in output.

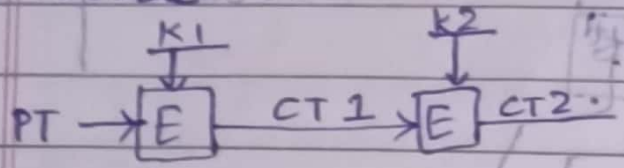
Plain Text 0000 . . . DES Algo . . . 0

Cipher Text . 4789 . . . . . JFAS

Completeness Effect : DES doesn't support 1 to 1 mapping

* Securities of DES
① Brute - Force Attack
② Differential Cryptanalysis
③ Linear Cryptanalysis

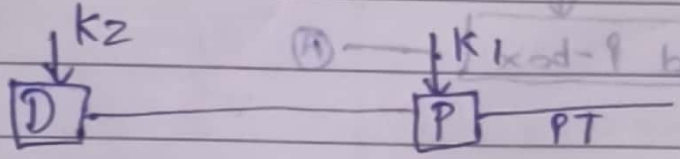
* Double DES



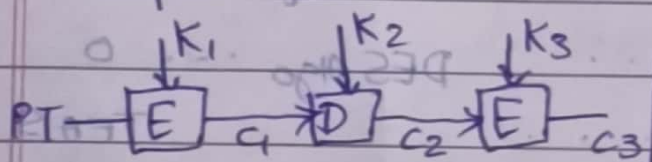
Encryption operation in double DES algo.

* Security of DES Algo - Read from Furozon

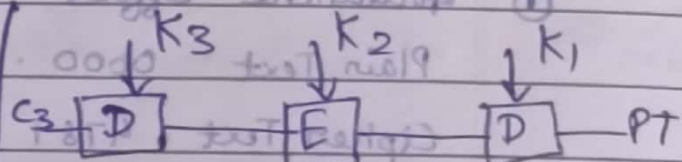
Rec



* Triple DES



Encryption



Decryption

1. $K_1 \neq K_2 \neq K_3 \rightarrow$ 168 bit
2. $K_1 = K_2 \neq K_3 \rightarrow$ 112 bit
3. $K_1 = K_2 = K_3 \rightarrow$ 56 bit

* Triple DES furozon

AES : Advanced Encryption Standard

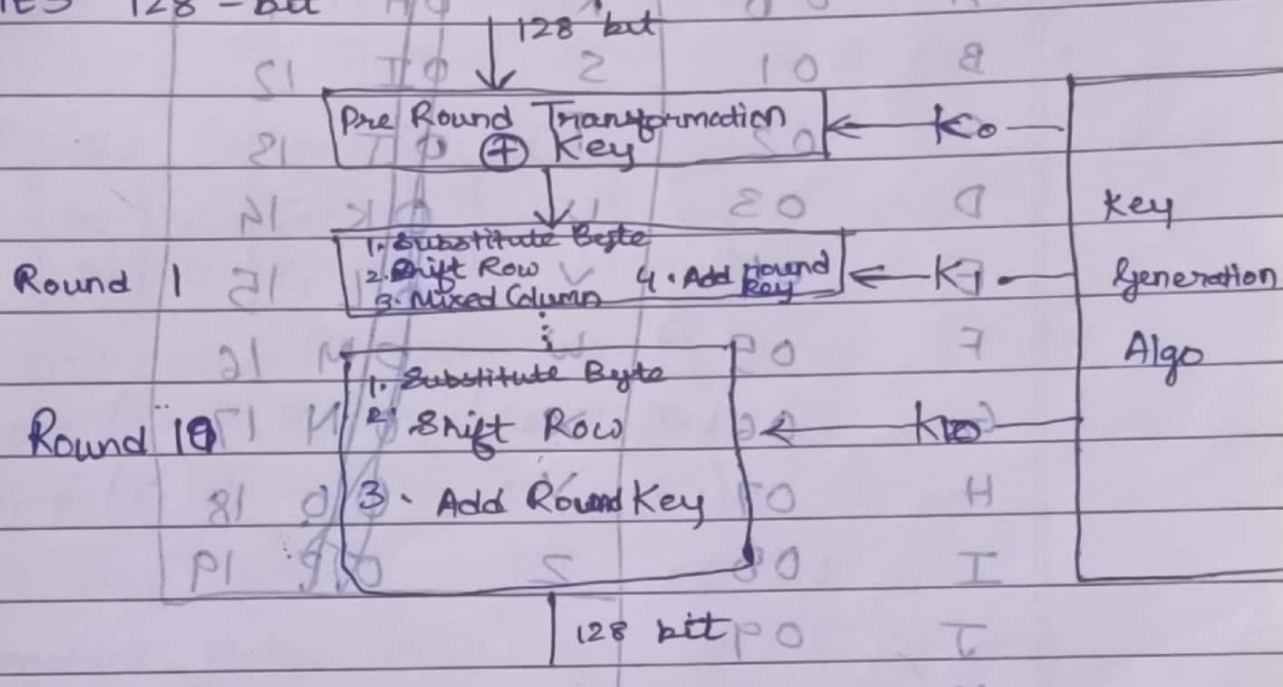
Invented in 2001 — NIST.

Aes used in Scandisk → FD – Flash Drive, HD – Hard Drive

iOS and ipAd OS → AES 128-bit Secured Access Program

	Key size	No. of Rond
① AES – 128 bit	128 bit	10
② AES 192	192 bit	12
③ AES 256	256 bit	14

* AES 128 – bit



Input state

a_0	a_4	a_8	a_{12}
a_1	a_5	a_9	a_{13}
a_2	a_6	a_{10}	a_{14}
a_3	a_7	a_{11}	a_{15}

Input ⇒ ABCDEF... OP

A	E	I	M
B	F	J	N
C	G	K	O
D	H	L	P

State 128 bit

PAGE No.
DATE / /

S- Box

2A	C1	3B	12
D3	F1	14	15

and

D3 14
F1 15

{ D3, F1, 14, 15 }

Shift Row Operation

4x4 initial state

63	47	a2	F0
F2	9C	63	65
F0	ab	7b	7c
af	76	76	Ca

⟹ After shift row

63	47	a2	F0
9C	63	65	F2
7b	7c	F0	ab
Ca	af	76	76

1 byte CLS
2 byte CLS
3 byte CLS

Mixed Column Operation

$$\begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} C_0 & C_2 \\ C_1 & C_3 \end{bmatrix} * \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$$

Const. C_1

$$b_0 = (C_0 * x_0) + (C_2 * x_1)$$

$$b_1 = (C_1 * x_0) + (C_3 * x_1)$$

Constant Matrix (4x4)

02	03	01	01
01	02	03	01
01	01	02	03
03	01	01	02

1 byte R.S.
1 byte R.S.
1 byte Circular R.S.

→ In exam Qs. will be : state 2×2, key - 2×2

Key : DJSANGHVICOECOMP

$$\begin{bmatrix} D & N & I & C \\ J & G & C & O \\ S & H & O & M \\ A & V & E & P \end{bmatrix}$$ 4×4 (key)

→ State value upper case me hi hoga, na ho to khud se upper case hi consider karna hai.

A	00	R	H	
B	01	S	I	12
C	02	T	J	13
D	03	U	K	14
E	04	V	L	15
F	05	W	M	16
G	06	X	N	17
H	07	Y	O	18
I	08	Z	P	19
J	09			
K	0A			
L	0B			
M	0C			
N	0D			
O	0E			
P	0F			

State ⊕ Key

A ⊕ D

= 00 ⊕ 03

= 00000000 ⊕ 00000011

= 00000011

= D

If const matrix is 2×2, then mtx. will always be

$$\begin{bmatrix} 02 & 03 \\ 01 & 02 \end{bmatrix}$$

Mix Column Operation

$$S = \begin{bmatrix} 63 & 47 \\ F2 & 9C \end{bmatrix} \quad C = \begin{bmatrix} 02 & 03 \\ 01 & 02 \end{bmatrix} \quad b = \begin{bmatrix} b_0 & b_2 \\ b_1 & b_3 \end{bmatrix}$$

Always while multiplying, write constant mtx. first.

$$b_0 = \begin{bmatrix} 02 & 03 \\ 01 & 02 \end{bmatrix} * \begin{bmatrix} 63 \\ F2 \end{bmatrix}$$

$$= \underbrace{(02 * 63) \oplus (03 * F2)}_{b_0} \oplus \underbrace{(01 * 63) \oplus (02 * F_2)}_{b_1}$$

Finite Field Arithmetic Operation

* In advanced AES algo., multiplication of 2 hexadecimal value is performed by using Finite Field arithmetic operation i.e. $G.F.(2^8) \rightarrow$ Galion field.
This operation is supporting max. scrambling/shuffling of data.

$b_0 = (02 * 63) \oplus (03 * F2)$

$02 = 0000\ 0010$

$63 = 0110\ 0011$

$GF(2^8) = x^7 + x^6 + x^5 + x^4 + \ldots x^1 + 1$

PAGE NO.
DATE / /

$GF(2^8) = \{0000\ 0000 \ldots 1111\ 1111\}$

$\{02\} = x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$

0 0 0 0 0 1 0

$= x$

$\{63\} = x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$

0 1 1 0 0 0 1 1

$= x^6 + x^5 + x + 1$

$(02 \times 63) = x * (x^6 + x^5 + x + 1)$

$= x^7 + x^6 + x^2 + x$

$= 1\ 1\ 0\ 0\ 0\ 1\ 1\ 0$

$= C\ 6$

$\{03\} = 0000 \oplus 0011 = x + 1$

$\{F2\} = 1\ 1\ 1\ 1 \quad 0\ 0\ 1\ 0 = x^7 + x^6 + x^5 + x^4 + x$

$(03 \times F2) = x^8 \oplus x^7 \oplus x^6 \oplus x^5 \oplus x^7 \oplus x^6 \oplus x^5 \oplus x^4 \oplus x$

$= x^8 + x^4 + x^2 + x$

Degree of above polynomial is 8 which is not a part of GF. Maximum degree supported by $GF(2^8)$ is 7. Now, we need to convert above poly into reduced poly. For that divide above poly by irreducible poly. $P(x) = x^8 + x^4 + x^3 + x + 1$

$P(x) = 100011011$

$t(x) = G_1(x) / P(x)$

$= 100010110$

```
100011011 | 100010110
          | 100011011
          | 00000 11011
```

$= 0D$

$(03 * F2) = 0D$

$b_0 = (02 * 63) \oplus (03 * F2)$

$= C6 \oplus 0D$

$= C0\,0D \oplus 0\,11\,01$

$= C\,1011 = CB$

$b_1 = (01 * 63) \oplus (02 * F2)$

$= 00000001 * 1111$

$= 1 * x^6 + x^5 + x + 1$

$= (x^6 + x^5 + x + 1) \oplus (x * x^7 + x^6 + x^5 + x^4$

$= (x^6 + x^5 + x + 1) \oplus (x^8 + x^7 + x^6 + x^5 + x^2)$

```
100011011 | 111100100
          | 100011011
          | 011111111
```

$= FF$

$b_1 = 63 \oplus FF$

$= 0110\ 0011 \oplus FF$

$= 0110\ 0011 \oplus 1111\ 1111$

$= 1001\ 1100$

$= 9C$

$b_2 = (02 * 47) \oplus (03 * 9C)$

$= (x * x^6 + x^2 + x + 1) \oplus (x+1 * x^7 + x^4 + x^3 + x^2)$

$= x^7 + x^3 + x^2 + x \oplus (x^8 + x^5 + x^4 + x^3 + x^7 + x^4 + x^3 + x^2)$

$$
\begin{array}{r}
1 \\
100011011 \overline{)\ 101001100} \\
\underline{100011011} \\
10111111 \\
\underline{10001101\ 1} \\
B\ F
\end{array}
$$

$03 \times 9C = BF$

$$b_3 = \begin{bmatrix} 02 & 03 \\ 01 & 02 \end{bmatrix} \begin{bmatrix} 47 \\ 9c \end{bmatrix}$$

$b_3 = (01 * 47) + (02 * 9C)$

$= (1 * x^6 + x^2 + x + 1)$

$= (0100\ 0111) + (x * x^7 + x^4 + x^3 + x^2)$

$= (0100\ 0111) + (x^8 + x^5 + x^4 + x^3)$

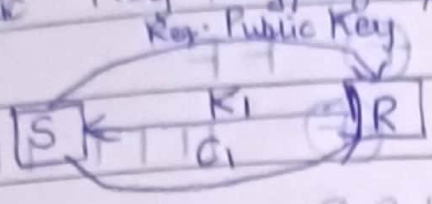
$= (0100\ 0111) + (0010\ 0011)$

$= 0110\ 0100$

$= 64$

RSA Algo

Public Key Cryptography



M ← Message (Plain Text)

K1 → Public Key
K2 → Private Key

Algo: ① Select 2 prime nos. p & q

② $n = p * q$

③ $z = \phi(n) = (p-1) * (q-1)$

④ $e = ?$

$e \xleftarrow{\text{Relative Prime}} z$

$g(e, z) = 1$

⑤ $d = ?$

$e * d \bmod n = 1$

⑥ Input value of plain text

$M = ?$

⑦ Encryption

$c = M^e \bmod n$

⑧ Decryption

$M = c^d \bmod n$

Diffie - Hellman Algo:

① Find primitive root of prime no.

P = 7 . (2, 3, 4, 5, 6)

$2^1 \bmod 7 = 2$

$2^2 \bmod 7 = 4$

$2^3 \bmod 7 = 1$

$2^4 \bmod 7 = 2$

$2^5 \bmod 7 = 4$

∴ 2 is not a primitive root of 7 bcoz 2 is repeated.

$3^1 \bmod 7 = 3$

$3^2 \bmod 7 = 2$

$3^3 \bmod 7 = 6$

$3^4 \bmod 7 = 4$

$3^5 \bmod 7 = 5$

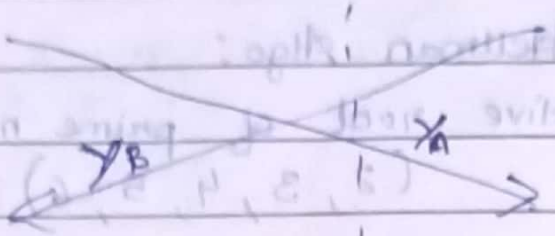
$3^6 \bmod 7 = 1$

α = ③ is a primitive root of 7.

Sender		Receiver
① Select ~~Primitive root~~ Prime no. P		
α → Primitive root of P.		P, α
	P, α →	
② Find out ~~key~~ private key (X_A) → Random value		② Find out private key (X_B)
③ Find out public key $y_a = (\alpha)^{X_A} \bmod P$		③ Find out public key $Y_B = (\alpha)^{X_B} \bmod P$

PAGE No.
DATE / /

(4)



(5) Secret key calculation

$y_1 = (y_B)^{X_A} \bmod P$.

→ Public key of receiver

(5) Secret key calculation

$y_2 = (y_A)^{X_B} \bmod P$

→ Public key of sender

$y_1 = y_2$

Eg. $P = 11$, $\alpha = ?$

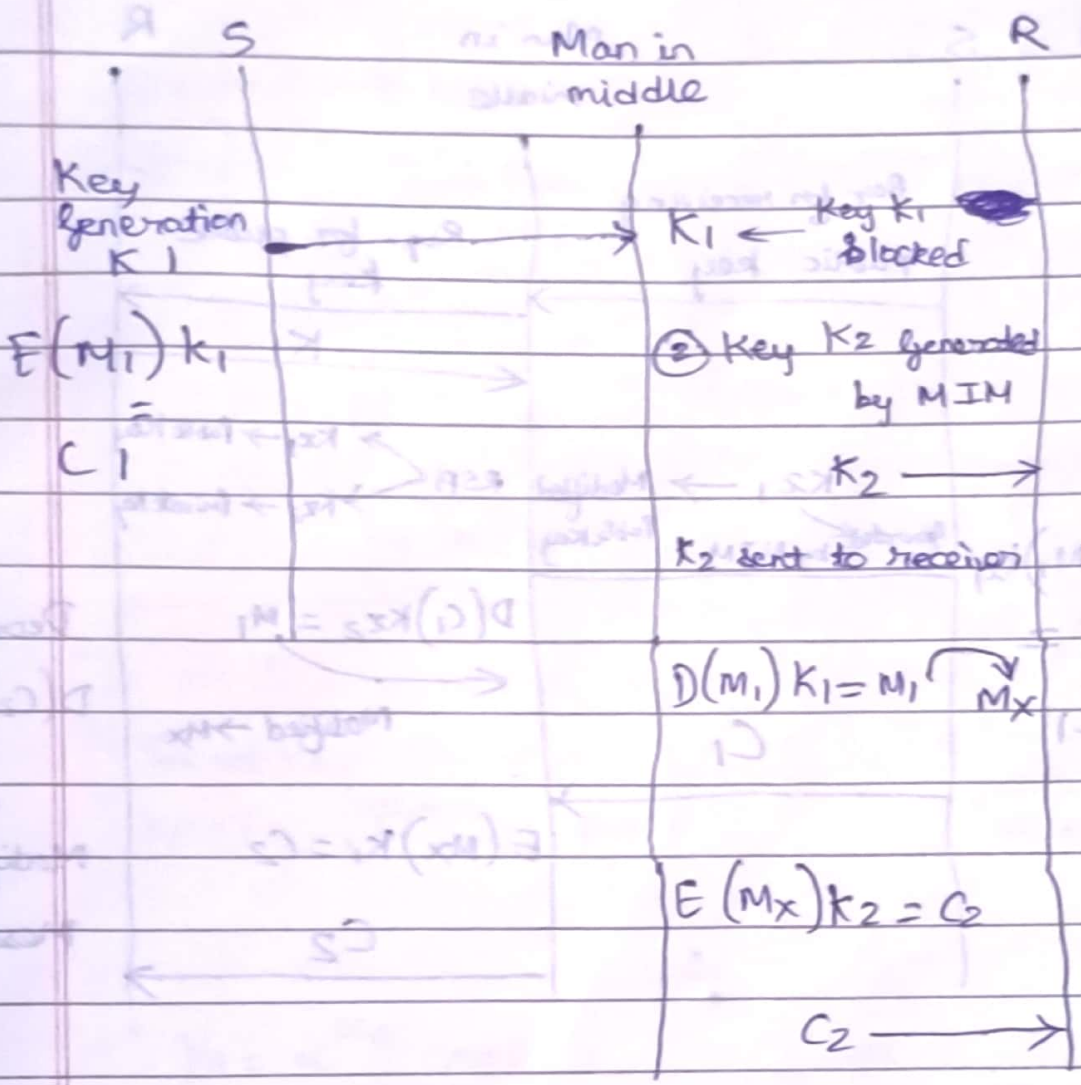
$2^1 \bmod 7 = 2$
$2^2 \bmod 7 = 4$
$2^3 \bmod 7 = 1$
$2^4 \bmod 7 = 2$
$2^5 \bmod 7 = 4$

$2^1 \bmod 11 = 2$
$2^2 \bmod 11 = 4$
$2^3 \bmod 11 = 8$
$2^4 \bmod 11 = 5$
$2^5 \bmod 11 = 10$
$2^6 \bmod 11 = 9$
$2^7 \bmod 11 = 7$
$2^8 \bmod 11 = 3$
$2^9 \bmod 11 = 6$
$2^{10} \bmod 11 = 1$

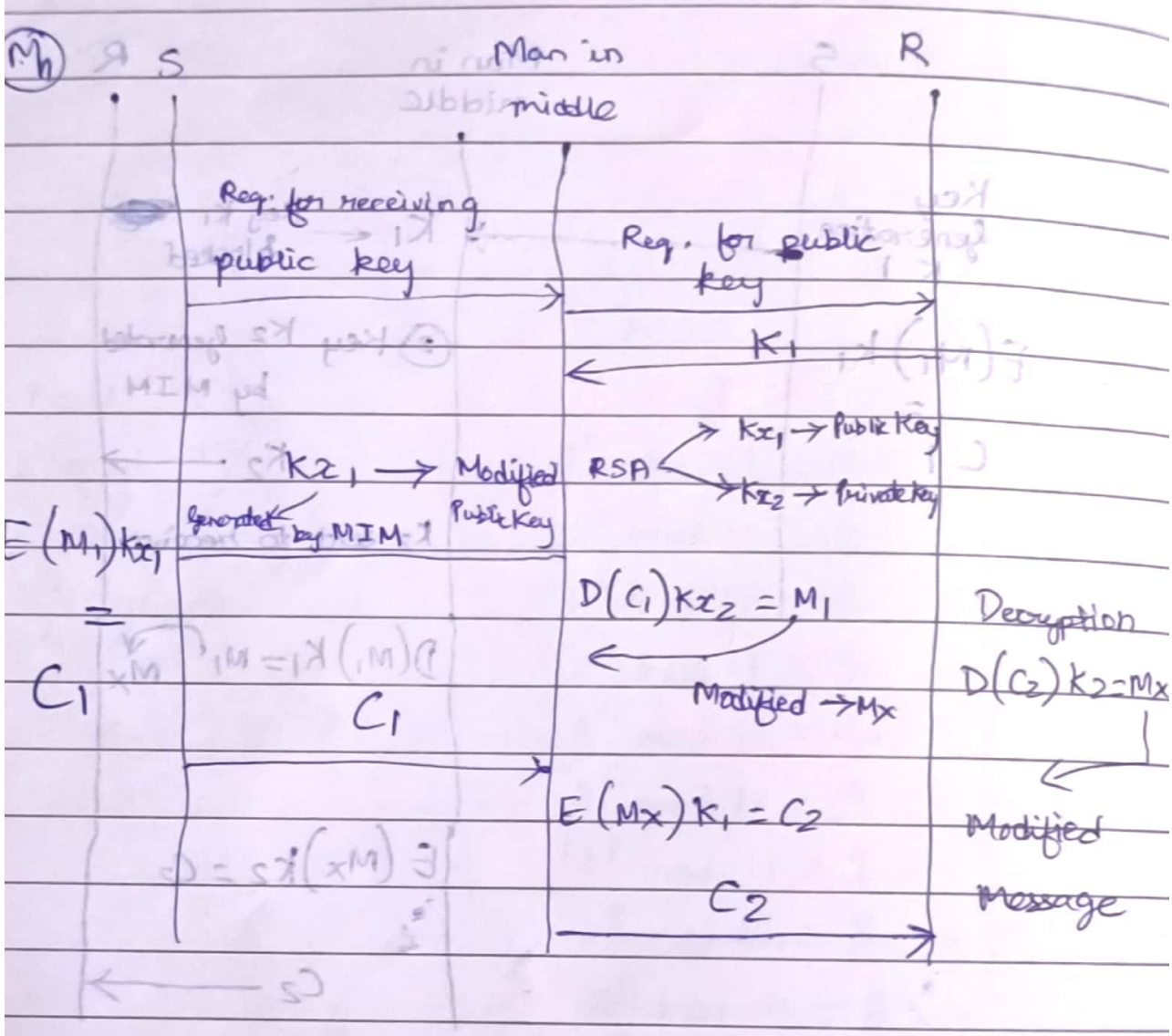
* Man - in - the - Middle Attack

- During Key Exchange Man in middle attack on

(a) Symmetric Key Exchange Algo.



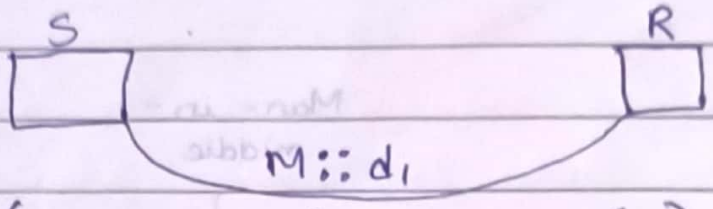
Man - in - Middle attack in Asymmetric Key Exchange



Digital Certificate and Authentication are the most important steps for countering MIM attack.

PAGE No.
DATE / /

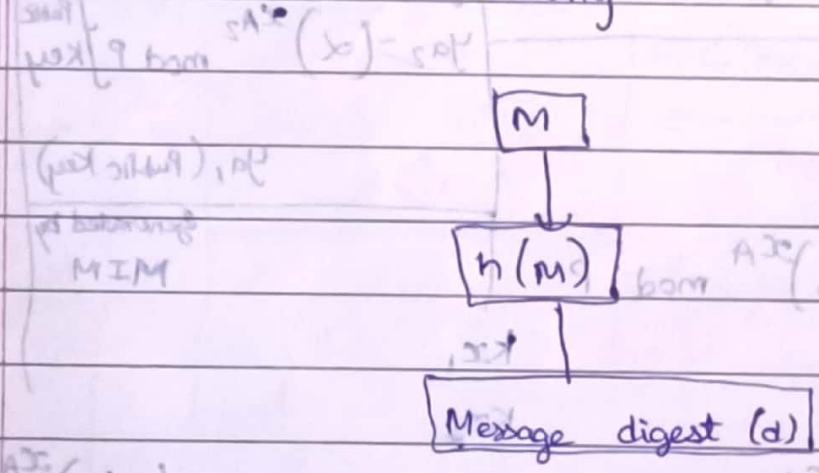
* Integrity



① $h(M) = d_1$ (Message digest) $\quad h(M) = d_2$

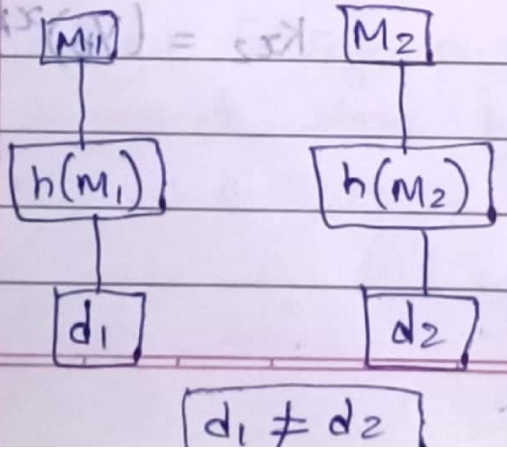
② M → Message $\quad d_2 - d_1 = 0$

Various hashing techniques are used to check the integrity of the message. Below diagram, is repra... how integrity of message is verified at receiving end.



① Non-reversible functio...

②



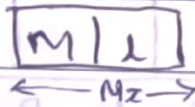
NOTE: Reading detail hashing algorithm from Furozen or Atul Kahate.

* Message Digest (MD5)
- Ron Rivest
- MD, MD2, MD3, MD4, MD5

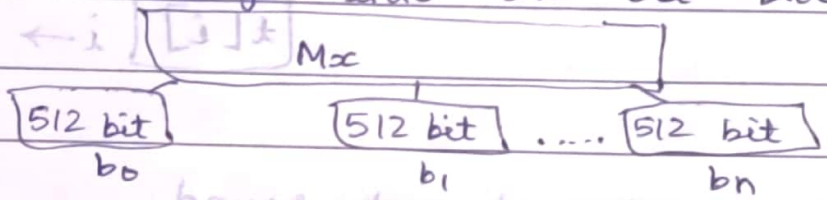
MD5 - Algo

① Calculate the length of message (l).

② Add length to the original message (M+l).



③ Divide message into 512 bit blocks.



④ Padding

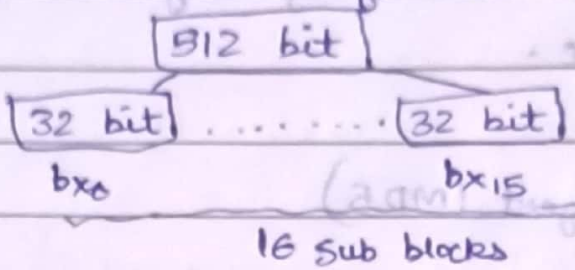
⑦ Divide 512 bit block into 16 sub block

⑤ Initialize chaining variable a, b, c, d. Size of each chaining variable is 32 bit.

⑥ Copy chaining variable into another temporary variable.

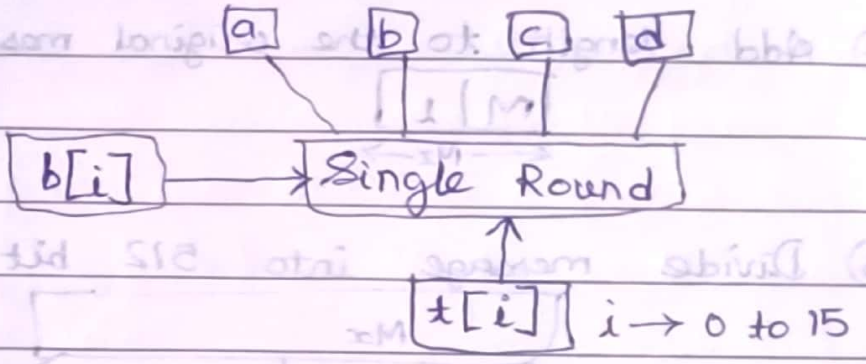
a	b	c	d	→ Chaining variable
a'	b'	c'	d'	→ Temp Chaining Variable.

(7) and each sub block of size 32 bit.

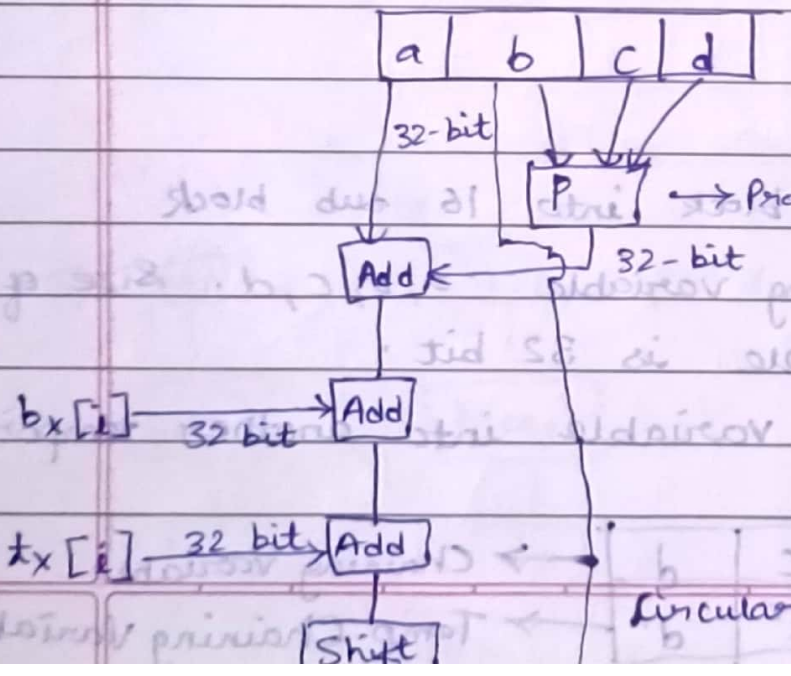


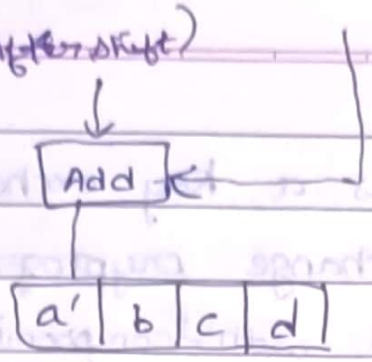
(8) MD5 algo. is performing 4 rounds of operation

Block diag. of each round is given as below.



Detail operation of each round





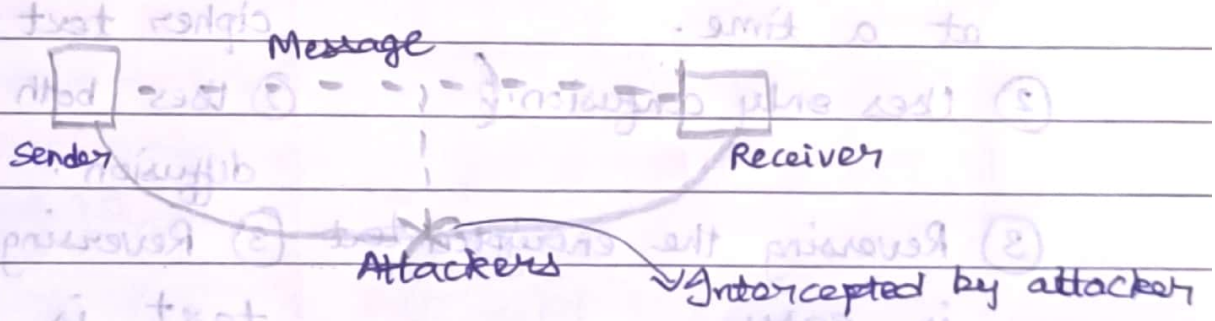
* Limitation

– Only single variable gets modified in each cycle

way, receiver will calculate his public key Y_b with help of his private key x_b.
$Y_b = \alpha^{x_b} \bmod P$. Then public keys are exchang
An attacker can intercept these public keys and manipulate them accordingly. This is not known to sender and receiver and the will continue to send messages to each other but it will be sent to attacker, and the attacker will manipulate the messages and send to both. This is how a MiM attack is carried out.

Digital certificates, digital signature and authentication are the steps that can be taken to enhance security against MiM attack.

* Integrity: A message sent from sender to receiver through a communication channel if not tampered or manipulated is called integrity



Integrity attacks aim to alter or forge data to deceive the recipient into accepting invalid or incorrect information.

PAGE No.
DATE / /

1. Few attacks on integrity are:

(i) Modification attack: It is an attack in which unauthorized alteration of data is done.

(ii) Replay attack: Replay attacks takes place when an attacker captures previous sequences of events, data and uses them later on to produce an unauthorized effect.

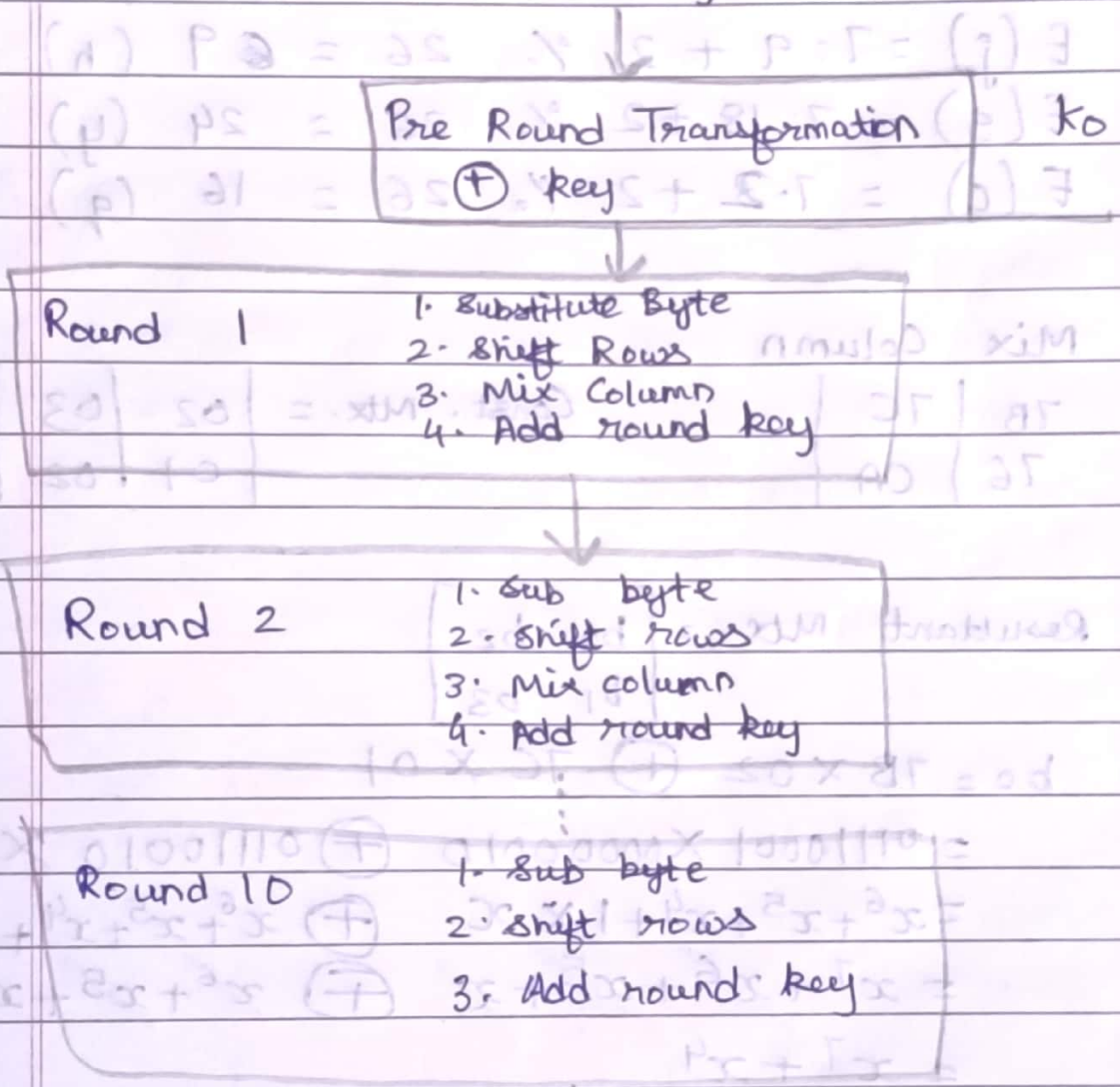
(iii) Masquerading: Pretending to be an authorized user to alter data.

(iv) Repudation: The sender or receiver can later deny to have sent or received the message.

* Stream Cipher Vs Block Ciphers

Stream	Block
(1) Converts plain text to cipher text by taking 1 bit plain text at a time.	(1) Plain text is divided into blocks and is converted to cipher text.
(2) Uses only confusion.	(2) Uses both confusion and diffusion.
(3) Reversing the encrypted text is easy.	(3) Reversing the encrypted text is hard.
(4) Complexity is hard.	(4) Complexity is simple.
(5) Works on substitution technique like caeser.	(5) Works on transposition techniqee like columnar.
(6) Eg: Caeser Cipher, RC4.	(6) Eg: DES, AES.

* AES 128 : AES stands for Advanced Encryption Standard. It is a type of block cipher. AES 128 is where the key size is of 128 bits. For size of 128 bit of key, there are 10 rounds. The following structure represents the flow of AES 128-bit.



Explain all operations with example.

* Affine Cipher :

Message = "djsc" key pair (7, 2)

d = 3 , j = 9 , s = 18 , c = 2

Encryption : $E = a \cdot x + b \% 26$

$E(d) = 7 \cdot 3 + 2 \% 26 = 23 \quad (x)$

$E(j) = 7 \cdot 9 + 2 \% 26 = 9 \quad (h)$

$E(s) = 7 \cdot 18 + 2 \% 26 = 24 \quad (y)$

$E(c) = 7 \cdot 2 + 2 \% 26 = 16 \quad (q)$

* Mix Column

7B	7C
76	CA

Const. Mtx. = $\begin{vmatrix} 02 & 03 \\ 01 & 02 \end{vmatrix}$

Resultant Mtx. = $\begin{bmatrix} b_0 & b_2 \\ b_1 & b_3 \end{bmatrix}$

$b_0 = 7B \times 02 \oplus 7C \times 01$

$= 01111001 \times 00000010 \oplus 01110010 \times 00000001$

$= x^6 + x^5 + x^4 + 1 \times x \oplus x^6 + x^5 + x^4 + x \times 1$

$= x^7 + x^6 + x^5 + x \oplus x^6 + x^5 + x^4 + x$

$= x^7 + x^4$

$= 10010000$

$=$

* Attacks on RSA signed digest.

- Key only attack: Three cases in this type

(i) Attacker intercepts the pair (S, M) [S: signature M: message] and tries to find another message m' that creates same digest, $h(m) = h(m')$. Since hash algos are second preimage resistant, this attack is very difficult.

(ii) Attacker finds two messages such that $h(m) = h(m$ Since hash algos are collision resistant, the attack is very difficult.

(iii) Attacker may randomly find message digest D, which may match with a random signature S. Attacker then finds message m such that $D = h(m)$ Since hash function is preimage resistant, the attack is very difficult.

- Known - Message Attack: Eve has 2 signature - message pairs (M_1, S_1) and (M_2, S_2) which have been created using same private key. Eve calculate $S \equiv S_1 \times S_2$. If she can find a message M such that $h(M) \equiv h(M_1) \times h(M_2)$, a new message has been forged.

- Chosen - Message Attack: Eve can ask Alice to sign two legitimate messages M_1 and M_2. Eve

then creates a new signature $S = S_1 \times S_2$. Since eve can calculate $h(M) = h(M_1) \times h(M_2)$ if she can find a message M given $h(M)$, the new message is a forgery.

ECC security and efficiency: ECC is a simulation of El Gamal cryptography. ① The security of ECC relies on solving the discrete logarithmic problem which is the difficulty of finding the integer k given points P and Q such that $Q = kP$. ECC provides good security even with keys much smaller in size compared to RSA, leading to faster computations and reduced power consumptions. ECC scales well with increasing key sizes, maintaining efficiency and security. Operations involved in ECC are less computationally intense, making it suitable for devices with limited processing power.

SHA-1: SHA-1 is a modified version of MD5 and so it closely resembles the structure of MD5. SHA-1 works for any text whose length is less than 2^{64}. The output message digest is of 160 bits. Operations involved in SHA-1 are:

Padding: Like MD5, padding is the first step in SHA-1 too where padding is added to end of original message in such a way that length of message is 64 bit short of multiple of 512.

Append Length: Length of message excluding the length of padding is appended to the end of padding as a 64-bit block.

Message is then divided into blocks each of 512 bits in length.

Initialize chaining variables: 5 chaining variables A to E each of length 32 bits are initialized. Values of A-D are same as in MD5 but E is initialized to Hex C3 D2 E1 F0.

Process Blocks: Copy chaining variables A-E into variables a-e. Then divide the current 512-bit block into 16 sub-blocks, each containing 32 bit